


[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent & Trademark Office

Try the *new* Portal design

Give us your opinion after using it.

Search Results

Search Results for: **[((try AND catch AND exception) AND (catch AND same AND (memory OR register) AND image))<AND>(meta_published_date <= 04-01-2000)<AND>(meta_path<CONTAINS>?)]**

Found **88** of **129,763** searched.

Search within Results

[> Advanced Search](#)[> Search Help/Tips](#)

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 88

short listing

**1****2****3****4****5****1** Implementing jalapeño in Java

82%



Bowen Alpern , C. R. Attanasio , Anthony Cocchi , Derek Lieber , Stephen Smith , Ton Ngo , John J. Barton , Susan Flynn Hummel , Janice C. Sheperd , Mark Mergen

ACM SIGPLAN Notices , Proceedings of the 14th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications October 1999

Volume 34 Issue 10

Jalapeño is a virtual machine for Java™ servers written in Java. A running Java program involves four layers of functionality: the user code, the virtual-machine, the operating system, and the hardware. By drawing the Java / non-Java boundary below the virtual machine rather than above it, Jalapeño reduces the boundary-crossing overhead and opens up more opportunities for optimization. To get Jalapeño started, a boot image of a ...

2 Java resources for computer science instruction

80%



Joseph Bergin , Thomas L. Naps , Constance G. Bland , Stephen J. Hartley , Mark A. Holliday , Pamela B. Lawhead , John Lewis , Myles F. McNally , Christopher H. Nevison , Cheng Ng , George J. Pothering , Tommi Teräsvirta

Working Group reports of the 3rd annual SIGCSE/SIGCUE ITiCSE conference on Integrating technology into computer science education December 1998

3 Java resources for computer science instruction

80%



Joseph Bergin , Thomas L. Naps , Constance G. Bland , Stephen J. Hartley , Mark A. Holliday , Pamela B. Lawhead , John Lewis , Myles F. McNally , Christopher H. Nevison , Cheng Ng , George J. Pothering , Tommi Teräsvirta

ACM SIGCSE Bulletin December 1998

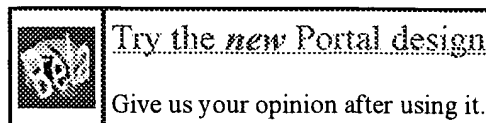
Volume 30 Issue 4

The goal of this working group was to collect, evaluate, and foster the development of resources to serve as components of both new and revised traditional courses that emphasize object-oriented software development using Java. These courses could, for



[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent & Trademark Office



Search Results

Search Results for: **[(*) <AND>((Kawahito OR Motohiro)<IN> author)]**

Found **8** of **129,763** searched.

Search within Results



[> Advanced Search](#)

[> Search Help/Tips](#)

Sort by: Title Publication Publication Date Score Binder

Results 1 - 8 of 8 short listing

- 1** Effectiveness of cross-platform optimizations for a java just-in-time compiler 82%



Kazuaki Ishizaki , Mikio Takeuchi , Kiyokuni Kawachiya , Toshio Suganuma , Osamu Gohda , Tatsushi Inagaki , Akira Koseki , Kazunori Ogata , Motohiro Kawahito , Toshiaki Yasue , Takeshi Ogasawara , Tamiya Onodera , Hideaki Komatsu , Toshio Nakatani
ACM SIGPLAN Notices , Proceedings of the 18th ACM SIGPLAN conference on Object-oriented programing, systems, languages, and applications October 2003
 Volume 38 Issue 11

This paper describes the system overview of our Java Just-In-Time (JIT) compiler, which is the basis for the latest production version of IBM Java JIT compiler that supports a diversity of processor architectures including both 32-bit and 64-bit modes, CISC, RISC, and VLIW architectures. In particular, we focus on the design and evaluation of the cross-platform optimizations that are common across different architectures. We studied the effectiveness of each optimization by selectively disabling ...

- 2** Effective sign extension elimination 82%



Motohiro Kawahito , Hideaki Komatsu , Toshio Nakatani
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2002 Conference on Programming language design and implementation May 2002
 Volume 37 Issue 5

Computer designs are shifting from 32-bit architectures to 64-bit architectures, while most of the programs available today are still designed for 32-bit architectures. Java™, for example, specifies the frequently used int" as a 32-bit data type. If such Java programs are executed on a 64-bit architecture, many 32-bit values must be sign-extended to 64-bit values for integer operations. This causes serious performance overhead. In this paper, we present a fast and effective algorithm for e ...

- 3** A dynamic optimization framework for a Java just-in-time compiler 82%

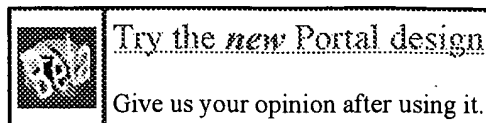


Toshio Suganuma , Toshiaki Yasue , Motohiro Kawahito , Hideaki Komatsu , Toshio Nakatani
ACM SIGPLAN Notices , Proceedings of the 16th ACM SIGPLAN conference on Object oriented programming, systems, languages, and applications October 2001
 Volume 36 Issue 11

The high performance implementation of Java Virtual Machines (JVM) and just-in-time


[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent & Trademark Office



Search Results

Search Results for: **[(*) <AND>((Ogasawara)<IN> author)]**Found **8** of **129,763** searched.

Search within Results

[> Advanced Search](#)[> Search Help/Tips](#)

Sort by: Title Publication Publication Date Score Binder

Results 1 - 8 of 8 short listing

- 1** A distributed, decision-theoretic control system for a mobile robot 82%
 Gary H. Ogasawara
ACM SIGART Bulletin July 1991
 Volume 2 Issue 4

- 2** Short Papers: Task specific eye movements understanding for a gaze- 80%
 sensitive dictionary
 Abdelaziz Khat, Yoshio Matsumoto, Tsukasa Ogasawara
Proceedings of the 9th international conference on Intelligent user interface
 January 2004
 In this paper, we study the relation between the user's degree of understanding and his/her eye movements; in an effort to realize a proactive interface that monitors the user and provides a contextual support. The application is a gaze sensitive dictionary that helps the user when reading a text in a browser's window. Not only is the user's gaze analyzed but also the context and thus the difficulty degree of the text being read. The experiment results suggest using regressions as an indicator t ...

- 3** Effectiveness of cross-platform optimizations for a java just-in-time 80%
 compiler
 Kazuaki Ishizaki, Mikio Takeuchi, Kiyokuni Kawachiya, Toshio Suganuma, Osamu Gohda, Tatsushi Inagaki, Akira Koseki, Kazunori Ogata, Motohiro Kawahito, Toshiaki Yasue, Takeshi Ogasawara, Tamiya Onodera, Hideaki Komatsu, Toshio Nakatani
ACM SIGPLAN Notices, Proceedings of the 18th ACM SIGPLAN conference on Object-oriented programing, systems, languages, and applications October 2003
 Volume 38 Issue 11
 This paper describes the system overview of our Java Just-In-Time (JIT) compiler, which is the basis for the latest production version of IBM Java JIT compiler that supports a diversity of processor architectures including both 32-bit and 64-bit modes, CISC, RISC, and VLIW architectures. In particular, we focus on the design and evaluation of the cross-platform optimizations that are common across different architectures. We studied the effectiveness of each optimization by selectively disabling ...

80%


[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent & Trademark Office

Try the *new* Portal design

Give us your opinion after using it.

Search Results

Search Results for: **[(*) <AND>((Komatsu)<IN> author)]**Found **20** of **129,763** searched.

Search within Results

[> Advanced Search](#)[> Search Help/Tips](#)

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 20 short listing

- 1** A very fast prolog compiler on multiple architectures 80%

Toshiaki Kurokawa , Naoyuki Tamura , Yasuo Asakawa , Hideaki Komatsu
Proceedings of 1986 fall joint computer conference on Fall joint computer conference November 1999
- 2** Posters: An evaluation of a method to detect and correct erroneous 80%

characters in Japanese input through an OCR using Markov models
 Tetsuo Araki , Satoru Ikehara , Nobuyuki Tsukahara , Yasunori Komatsu
Proceedings of the fourth conference on Applied natural language processing
 October 1994
 The "*Selective Error Correction Method*" to judge these three types of errors, and correct them, using *m*-th order Markov chain model for Japanese 'kanji-kana' characters, has been proposed and shown to be useful to detect and correct errors generated randomly (Araki et al., 1994). In this paper, this method is applied to detect and correct erroneous characters in Japanese text input through an OCR. The method is confirmed to be also effective to detect and correct the errors introduc ...
- 3** Effectiveness of cross-platform optimizations for a java just-in-time 80%

compiler
 Kazuaki Ishizaki , Mikio Takeuchi , Kiyokuni Kawachiya , Toshio Suganuma , Osamu Gohda , Tatsushi Inagaki , Akira Koseki , Kazunori Ogata , Motohiro Kawahito , Toshiaki Yasue , Takeshi Ogasawara , Tamiya Onodera , Hideaki Komatsu , Toshio Nakatani
ACM SIGPLAN Notices , Proceedings of the 18th ACM SIGPLAN conference on Object-oriented programing, systems, languages, and applications October 2003
 Volume 38 Issue 11
 This paper describes the system overview of our Java Just-In-Time (JIT) compiler, which is the basis for the latest production version of IBM Java JIT compiler that supports a diversity of processor architectures including both 32-bit and 64-bit modes, CISC, RISC, and VLIW architectures. In particular, we focus on the design and evaluation of the cross-platform optimizations that are common across different architectures. We studied the effectiveness of each optimization by selectively disabling ...

80%